Towards robot cultures?

Learning to imitate in a robotic arm test-bed with dissimilarly embodied agents

Aris Alissandrakis, Chrystopher L. Nehaniv and Kerstin Dautenhahn
University of Hertfordshire, United Kingdom

The study of imitation and other mechanisms of social learning is an exciting area of research for all those interested in understanding the origin and the nature of animal learning in a *social context.* Moreover, imitation is an increasingly important research topic in Artificial Intelligence and social robotics which opens up the possibility of *individualized social intelligence* in robots that are part of a community, and allows us to harness not only individual learning by the single robot, but also the acquisition of new skills by observing other members of the community (robots, humans, or virtual agents). After an introduction to the main research issues in research on imitation in various fields (including psychology, biology and robotics), we motivate the particular focus of this work, namely the *correspondence problem.* We describe *Action Learning for Imitation via Correspondences between Embodiments* (Alice), an implemented generic framework for solving the correspondence problem between differently embodied robots. Alice enables a robotic agent to learn a behavioral repertoire suitable to performing a task by observing a model agent. Importantly, the model agent could possibly possess a different type of body, e.g. a different number of limbs or joints (implying different degrees of freedom), a different height, different sensors, a different basic action repertoire, etc. Previously, in a test-bed where the agents differed according to their possible movement patterns, we demonstrated that the character of imitation achieved will depend on the granularity of subgoal matching, and on the metrics used to evaluate success.

In our current work, we implemented Alice in a new test-bed called *Rabit* where simple simulated robotic arm agents use various metrics for evaluating success according to actions, states, effects, or weighted combinations.

We examine the roles of *synchronization*, *looseness of perceptual matching*, and of *proprioceptive matching* by a series of experiments. Also, we study how

Interaction Studies 5:1 (2004), 3–44.
issn 1572–0373 / e-issn 1572–0381

Alice copes with *changes in the embodiment* of the imitator during learning. Our simulation results suggest that *synchronization* and *loose perceptual matching* allow for faster acquisition of behavioral compentencies at low error rates.

Social learning (broadly construed) plays a role as a *replication* mechanism for behaviors and results in *variability* when the transmitted behavior differs from the model's behavior, thus providing the *evolutionary substrate for culture* and its pre-cursors. Social learning in robotics could therefore serve as the basis for culture in societies whose members include artificial agents. We address the use of imitative social learning mechanisms like Alice for transmission of skills between robots, and give first examples of transmission of a skill despite differences in embodiment of agents involved. In the particular setup, transmission occurs through a chain, as well as emerging in cyclic arrangements of robots. These simple examples demonstrate that by using social learning and imitation, *(proto-)cultural transmission* is possible among robots, even in heterogeneous groups of robots.

1. Introduction: Acquiring skills via social learning by imitation

Imitation is an important means to acquire new competencies in a social context. Human cultures use imitation in a variety of ways. This may include learning of new motor skills, such as learning how to tie shoe laces, how to play tennis, or learning a dance. Imitation may also be involved when we learn a language, learn tool use, or learn how to behave in particular social contexts, such as learning dress codes or greetings. In animal sciences the question of which animal species imitate is under hot debate. Imitation is different from other mechanisms of social learning, e.g. when animals learn mainly due to the *presence* of conspecifics, or when conspecifics draw attention to certain *features* of the environment that are involved in the behavior, which the animal then acquires through further individual (non-social) learning. In the latter case biologists would call this form of social learning *local* or *stimulus enhancement*. Other types of social learning (also called observational learning) are variously taxonomized and include *goal emulation*, *social learning of affordances*, etc.

Given the controversial nature of definitions of imitation discussed in biology and psychology, how can we define imitation in the context of our work on robot learning? What existing definitions are most appropriate? Thorndike's well known definition (Thorndike 1898) defined imitation as any situation in which animals "from an act witnessed learn to do an act". Importantly, Thorndike's definition requires that imitation involves *learning*. This means that his definition excludes instances where no learning is involved. Later, in 1963, Thorpe defined 'true imitation' as the "copying of a novel or otherwise improbable act or utterance, or some act for which there is clearly no instinctive tendency" (Thorpe 1963). The issue of how to define imitation is still controversial in the animal behavior literature. Some authors, cf. Thorpe's definition, require that 'do as I do' (the common sense interpretation of imitation or 'apeing') should only be called imitation if it involves learning a *novel* behavior (Visalberghi & Fragaszy 2002), which excludes instances when an animal copies the movements/behavior of another animal when the movements or the behavior were already in the (innate or learned) repertoire of the animal. See (Zentall 2001, Dautenhahn & Nehaniv 2002, and Dautenhahn, Nehaniv & Alissandrakis 2003) for an in depth discussion of different types of social learning, definitions of imitation, and the main research questions involved.

In the context of our work on robot imitation we adopt the following useful definition due to Mitchell (1987) since it is applicable across animals and machines and therefore suitable for an application involving (simulated) robots. According to Mitchell, *imitation* occurs when

– something *C* (the copy) is produced by an organism or machine, where
– *C* is similar to something else *M* (the model)
– registration (or perception) of *M* is necessary for the production of *C*, and
– *C* is designed to be similar to *M*.

In this definition 'design' can refer to design by an agent, by nature, or by evolution (e.g. when an organism selects and performs a behavior to closely match that of another agent in some aspect; or in the sense that nature 'designed' the scarlet king snake to mimick the appearance of the poisonous coral snake (Pfennig, Harcombe & Pfenning 2001)); or design by humans (e.g. the photocopier designed to produce an exact copy). This definition is very attractive since it is broad and encompasses a number of instances where we can observe similarity between demonstrator (model) and imitator. Note that this definition does not demand novelty of the copy *C*. Fragaszy & Visalberghi (1990) later added such a novelty requirement to Mitchell's definition of imitation, since they were primarily interested in instances of clear evidence for whether or not certain animals have a capacity for imitation. In the context of our experimental framework, where novelty can be obtained easily[1] and is hence not at issue, we will use the term 'imitation' according to Mitchell's original definition. Finally, it is important to note that Mitchell's requirements that C should be *similar* to M and designed to be similar to M tacitly use undefined notions of similarity. These requirements might be realized by — or the degree to which they are met might be evaluated by — using *metrics*, i.e. quantitative measures of similarity. Also different metrics of similarity could be used in achieving or evaluating qualitatively different types of imitation; and, moreover, different metrics could be used together for each of these two requirements in a given case of imitation.

2. Research on imitation in robotics

Roboticists have been interested in imitation since the early 1990's (Kuniyoshi, Inoue & Inaba 1990, Dautenhahn 1994). In the robotics community the term 'imitation' is often used in a much broader sense than in the animal literature. The primary goal of most computer scientists and roboticists is to create artifacts that show certain skills or behaviors (e.g. that can learn by demonstration from a human), which is very different from the possible interest of a biologist in understanding the exact nature and the ecological or evolutionary context of a certain behavior in relation to other naturally occurring behaviors in a given species or related species. Robot imitation 'in a broader sense' might refer to copying, mirroring, or other types of 'imitative' behavior. For example, *learning by imitation* involving a following strategy (matched dependent behavior of the imitator following the model around in an environment) has been used widely for facilitating robot learning (Demiris & Hayes 1996, Billard & Dautenhahn 1998).

Robotics researchers are often inspired by biology when creating controllers for their autonomous robots, using suitable behaviors for adaptive learning (Kuniyoshi, Inoue & Inaba 1990, Demiris & Hayes 1996, Billard 1998, Gaussier, Moga, Banquet & Quoy 1998, Schaal 1999, Matarić, Jenkins, Fod & Zordan 2000). Imitation is a powerful learning tool when social interaction either between human and robots or even in multi-robot systems is involved. Having a robot observe and learn to perform a task from an experienced teacher presents a more flexible and adaptive solution than explicit programming of robot behavior. The learning process can be faster as no direct teaching is required: the expert should, just by performing and thus demonstrating the task, pass the required knowledge on to the robot, which in turn may be used as a model to be imitated by other robots (cf. experiments on learner robots that can become teachers in (Billard & Dautenhahn 1998)).

Imitation can have another important role in robotics, besides skill acquisition, namely it might serve as a stepping stone towards the development of the kinds of social cognition found in humans, and possibly other animals. For example, developmental psychologists have revealed the crucial role of imitation in how humans become social beings, e.g. how they identify others as persons, and how they recognize individuals (cf. work on neonatal imitation, e.g. Meltzoff 1996, Nadel & Butterworth 1999 and others). It has been proposed that imitation could be used in a similar role for developing artificial social intelligence in robots and other artificial agents (Dautenhahn 1994, 1995, Scassellati 1999). But robotics research in imitation seems easily biased towards divorcing the imitative *mechanism* from the social dimension of imitation, thus ignoring the potential of social integration for robotic agents with other artificial agents and humans. Traditionally, robotics research aims at developing architectures that (usually using a vision system) identify salient features in the movements/actions of a model agent and map them to appropriate motor outputs of the robot imitator (Kuniyoshi, Inoue & Inaba 1990, Kuniyoshi, Inaba & Inoue 1994). Focusing on the question of how to imitate *given a particular robotic system and a specified task*, leads to very diverse control approaches that are often difficult to generalize across different platforms and contexts.

Overall, there seems to exist a lack of a *general* framework for *how to imitate*, and this is the main issue addressed in our approach. This approach is very different from the well-studied direction of *learning by imitation* which assumes that an artifact already possesses the skill to imitate successfully and in turn exploits this ability as a means to acquire knowledge, cf. the above discussion of following strategies (Hayes & Demiris 1994, Billard & Dautenhahn 1998, Billard, Hayes & Dautenhahn 1999).

More recently, control architectures for robots that can imitate have greatly benefitted from neurobiological inspiration, in particular from findings on the mirror-neuron system, e.g. Gallese, Fadiga, Fogassi & Rizzolatti (1996). Examples of such neurobiologically-inspired research can be found in Demiris (1999), Demiris & Hayes (2002), and Billard & Schaal (2001) where software avatars were used instead of physical robots. In Demiris (1999), Demiris & Hayes (2002) an imitator avatar learns the international standard semaphore code (ISSC), a set of actions representing movements from the rest position to specific postures, corresponding to letters of the alphabet. The control architecture developed allows study of the acquisition of both single letters, as well as sequences of letters forming words. Billard & Schaal (2001) present a biologically inspired model of human imitation, implemented in a dynamic simulation of a humanoid avatar, reproducing human arm motion. The model learns the principal features of a 3 degrees-of-freedom (DOF) trajectory using captured data of human arm motion, and generalizes across the different demonstrations. The control architectures developed in this work can serve as models to both understand the neurological mechanisms underlying imitation and also help develop further robot architectures and controllers that can imitate.

Generally, the acquisition of motor skills is a very active area of research, involving learning control policies to match via-points in the motion trajectory of a model (Schaal 1999) or, for example, in work where a virtual humanoid agent learns to dance the Macarena (Matarić, Zordan & Mason 1998). Examples of research in learning motor skills by observation are described in (Bentivegna & Atkeson 2002), involving a test-bed where a humanoid robot learns how to play air hockey from a human model.

Note, as has been argued elsewhere (e.g. Dautenhahn & Nehaniv 2002, Breazeal & Scassellati 2002), although 'imitation' is often used in robotics without very precise definitions, robotics addresses the challenge to precisely define, synthesize and experimentally test *mechanisms* and *procedures* of perception, action and learning involved in imitative learning. In other words, a theory as such cannot make a robot imitate, but a control program (informed by a theory) possibly can. Thus, imitation is an area of research that truly demands mutually fruitful, interdisciplinary collaborations that can shed light on understanding why and how humans and other animals imitate, and how artifacts can be synthesized harnessing imitative behavior for skill learning, social, communicative, or other purposes, possibly even providing a path towards robots with human-like characteristics (Schaal 1999).

3. The correspondence problem

While much previous work in Artificial Intelligence and robotics has engineered *ad hoc* mechanisms to achieve imitation, general mechanisms for solving the *correspondence problem*, i.e. *how an imitating agent can imitate a model with possibly dissimilar embodiment, mapping its perceptions of the model's behavior to its own actions*, are our focus in this paper. A semi-formal definition of the *correspondence problem* (Nehaniv & Dautenhahn 1998, 2000, 2001, 2002) is as follows:

> Given an observed behavior of the model, which, from a given starting state, leads the model through a sequence (or hierarchy [or program]) of subgoals (in states, action, and/or effects, while possibly responding to sensory stimuli

> and external events), find and execute a sequence of actions using one's own (possibly dissimilar) embodiment, which, from a corresponding starting state, leads through corresponding subgoals (in corresponding states, actions, and/or effects, while possibly responding to corresponding events). (Nehaniv & Dautenhahn 2002: 49)[2]

The notion of 'corresponding' states, actions and effects here can be formalized by a choice of metrics, and the choice of subgoals to be matched defines the granularity and program-structure of the candidate matching behavior (Nehaniv & Dautenhahn 1998). Our previous work studying the correspondence problem included a *Chessworld* scenario where the results showed that the newly developed framework called *Action Learning for Imitation via Correspondences between Embodiments* (ALICE) could solve the correspondence problem for agents with dissimilar embodiments (Alissandrakis, Nehaniv & Dautenhahn 2001, 2002). The *Chessworld* scenario consisted of a chessboard on which chess pieces (dissimilarly embodied agents) moved around according to rules defined by their type, e.g. the Bishop can only move diagonally on squares of the same colour, the King can only move one square per move, etc. The intention was to only borrow elements like the simple discrete nature and the different piece embodiments (with different permitted movement patterns constraining their possible actions in the ambient environment); we were not interested in agents playing the actual game of chess nor in the physical appearance/shape of the chess pieces. In *Chessworld*, each move resulted in a displacement of the piece on the chessboard, and these actions were to be imitated by other chess pieces. Results from this previous work also demonstrated that the metric and the level of subgoal granularity can each dramatically affect the character of imitative behavior that is generated, and that one metric is not in general universally 'better' than another, but various choices of metrics contribute to which aspects of a behavior are to be matched in imitation (Alissandrakis, Nehaniv & Dautenhahn 2002).

In our current work we use a different test-bed, namely a scenario where robotic arms of different types imitate each other.

4. Immediate imitation: Synchrony and social intelligence

As we already mentioned above in Section 2, imitation can have another important role in robotics, besides skill acquisition. Developmental psychologists have revealed the crucial role of imitation in how humans become social beings, e.g. how they identify others as persons who are 'like-me', and how they recognize individuals (cf. work on neonatal imitation, e.g. Meltzoff 1996, Butterworth 1999 and others).

Synchronization of behavior plays a fundamental role in child-caretaker interactions, as becomes evident in developmental studies with babies and infants, e.g. (Trevarthen, Kokkinaki & Friamenghi, Jr. 1999, Nadel, Guerini, Peze & Rivet 1999). The contingent and dynamic aspects of interaction and communication are stepping stones in the social development of infants, and they are prerequisites of *immediate imitation* when imitation is shown very soon after observing the model actions, as opposed to deferred imitation where the observing agent imitates the model with a significant temporal delay without any immediate practice. *Synchronic imitation* is closely related to immediate imitation. The former puts emphasis on model and imitator carrying out a behavior synchronously, in concert with each other, cf. definitions used in Nielsen & Dissanayake (2003) where a longitudinal study with infants showed that while immediate and deferred imitation could be observed from 12 months onwards along a similar developmental trajectory, synchronic imitation only emerged after 18 months, which, according to the investigators, suggests that synchronic imitation is primarily a communicative behavior. Note that synchronic and/or immediate imitation can serve important functions in intentional communication, involving, e.g., role switching, monitoring of synchrony, sharing of topics and emotions, see a detailed discussion in Nadel (2002).

It has also been argued by Nadel, Guerini, Peze & Rivet (1999) that immediate imitation creates intersubjectivity and is the first step by which infants make 'contact' to other human beings. Individualized social intelligence in humans and social animals may rely on a common core of these and related mechanisms (cf. Whiten & Byrne 1988, Pepperberg 2002). This and other evidence from the study of animal social complexity (e.g. Herman 2002) suggest that synchronization and immediate imitation might also be key ingredients for the development of *individualized social intelligence in robots* (Dautenhahn 1994, 1995). We show below that use of synchronization of behavior in a robotic test-bed has another possible benefit: it can also significantly speed up the solution of the correspondence problem.

5. The Alice framework in a robotic arm test-bed

In order to study the correspondence problem we developed the Alice (Action Learning via Imitation between Corresponding Embodiments) generic framework for imitation. This framework provides the architecture of a controller for the actions of an imitating agent, making use of a *correspondence library*. The *keys* to the entries of this library consist of some defined combination of actions/states/effects of the model agent, and proprioceptive information concerning the imitator's own state. Perceptions of the model and possibly proprioceptions are converted to the form of a key for the imitator to look-up corresponding matching actions (or action sequences) for that key in the imitator's correspondence library. These are the actions that the imitating agent should perform in order to achieve a matching behavior, according to an evaluation metric. As better matching actions corresponding to the perceptual keys are learned they are added to the imitator's library (which is initially empty). The correspondence library thus consists of re-useable mappings between keys encoding what the model does and action sequences which the imitator has learned to use in attempting to match it. In the library, a key may be associated with several possible matching actions (or action sequences). A schematic view of the Alice framework showing major components appears in Figure 1.

Candidate matching actions can be generated using any kind of generating algorithm to propose actions, e.g. inverse kinematics where given the end point of a robotic arm the corresponding joint angles needed to achieve that end point are computed. In the work here we simply use a *random* generating algorithm, since we are not concerned about the precise nature of the generating mechanism here and would like to distinguish the effectiveness of Alice from success that might be merely due to the use of a sophisticated generating

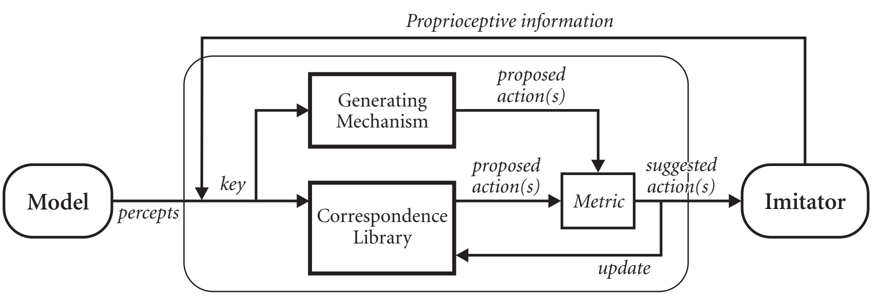

Figure 1. The ALICE framework. *Percepts* of imitator arising from the model's behavior (actions, states and effects) and *proprioceptive information* (own state) of the imitator form components of a *key* that is used by the *correspondence library* (if it matches any of the existing entry keys at that stage of the library's growth) and by the *generating mechanism* to produce a sequence of one or more *proposed action(s)*. These are evaluated using a *metric*, and the correspondence library is updated accordingly with the resulting *suggested action(s)* for the imitator.

mechanism. Proposed actions are then evaluated according to a metric and will either update an existing entry with more fitting solutions, or create a new entry of their own if the current state/action/effect or proprioceptive aspects comprising the key have not been previously observed.

The type of the resulting imitating behavior will depend on the granularity and selection of percepts to match, and on the metric used, e.g. whether the imitator will try to match the perceived model actions/states/effects or some combination of them. For more details on Alice see Alissandrakis, Nehaniv & Dautenhahn (2002), Alissandrakis (2003) and below.

5.1 Related work in ethology and psychology

Several theories of imitation are suggested in ethology and psychology. Our particular generic framework for solving the correspondence problem in learning how to imitate is related, e.g., to statistical string-parsing models of social learning from ethology (Byrne 1999) and the *associative sequence learning theory (ASL)* from psychology (Heyes & Ray 2000). Space only permits discussion of ASL which we consider more closely relevant to the Alice framework presented in this paper.

Heyes and Ray's ASL theory (Heyes & Ray 2000, Heyes 2002) (see Figure 2) is explicit enough to make testable predictions in the experimental psychology of humans and animals. The theory assumes that a *sequence of action units* may compose a behavior to be imitated rather than its being unitary. The resolution of these action units can vary, depending on the observer's perception. In order for an observer to imitate a sequence, ASL requires two processes to successfully take place. The *horizontal process* associates the mental representations of these action units in a successive chain, so that the observer knows what the sequence 'looks like'. The *vertical process* associates, directly or indirectly, each of the sensory representations of the action units to appropriate motor representations, so that the observer can perform the sequence.

In the Alice framework, the vertical associations are captured by a *correspondence library* (see above), while the temporal sequence of actions by the model and imitator agents relates to the horizontal processes of ASL. The primary purpose of the ASL framework is to stimulate the development of other testable models of imitation and guide analytic psychological experiments. One of the strengths of ASL is that it can also be applied to *perceptually opaque* actions like facial expressions, actions that yield dissimilar sensory inputs to an agent when observed and when executed. One of the weak points of the current

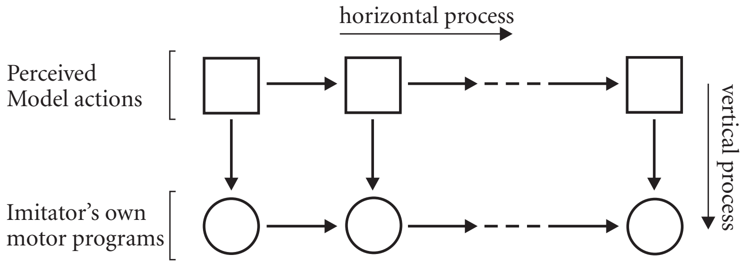

Figure 2. Associative Sequence Learning (ASL) theory. Two processes are required to successfully take place in imitation. The behavior of the model is broken down into a sequence of elementary action units. The *horizontal process* associates the mental representations of the action units of the model in a successive chain. The *vertical process* associates directly or indirectly each of the sensory representations to appropriate motor representations. The combination of both processes allows the imitator to perform an action sequence matching the perceived model actions. (Figure inspired from (Heyes 2002).)

ASL framework is that it does not seem to address the state of the agents' bodies or the effects of actions on the environment at all. The Alice framework described here allows both states and effects to be handled along with actions, using appropriate metrics.

6. The *Rabit* test-bed

The *R*obotic *A*rm em*B*odiment for *I*mitation *T*estbed (*Rabit*) was created as a simple, yet rich enough, environment to allow for several dissimilarly embodied model and imitator agents, with embodiments with differing numbers of degrees of freedom, to be considered, in order to evaluate the Alice framework.[3]

In *Rabit* model and imitator agents are implemented as robotic arms that can have any number and length of segments, drawing shapes on a two-dimensional workspace. A *Rabit* model behavior is perceived by the imitator as a sequence of *actions* (rotation of the joints) that result in a change of *state* (arm posture) and can have *effects* (a continuous trail of 'paint' on the workspace, left behind by the end tip of the arm) on the environment. For each of these aspects, percepts can be made available to the imitator and a separate metric can be used to compare between the model and the imitator. Thus both model and imitator agents are embodied as robotic arms that can have any number of rotary joints, each segment of varying length, occupying a two-dimensional workspace (see Figure 3). The agent embodiment can be described as the vector $L=[l_1, l_2, l_3 \ldots, l_n]$, where l_i is the length of the ith segment of the robotic arm.

There is no modelling of complex physics in the workspace and the movement of the arms is simulated using simple forward kinematics but without collision detection or any static restraints (in other words, the segments of the arm can bend into each other). The intention is to explore the features of the ALICE framework, not to build a faithful robotic arm simulator.

6.1 Aspects of the correspondence problem: Actions, states, effects

To study the aspects of matching actions, states, and effects, in the correspondence problem, we must define these terms for the robotic-arm test-bed:

An *action* of a given agent is defined as a vector describing the change of angle for each of the joints, $A=[\alpha_1, \alpha_2, \alpha_3, \ldots, \alpha_n]$, where n is the number of its joints. These angles are relative to the previous state of the arm and can only have three possible values, +10° (anti-clockwise), 0° or −10° (clockwise).

A *state* of an agent is defined as the absolute angle for each of the joints, $S=[\sigma_1, \sigma_2, \sigma_3, \ldots, \sigma_n]$, where n is the number of its joints. A distinction can be made

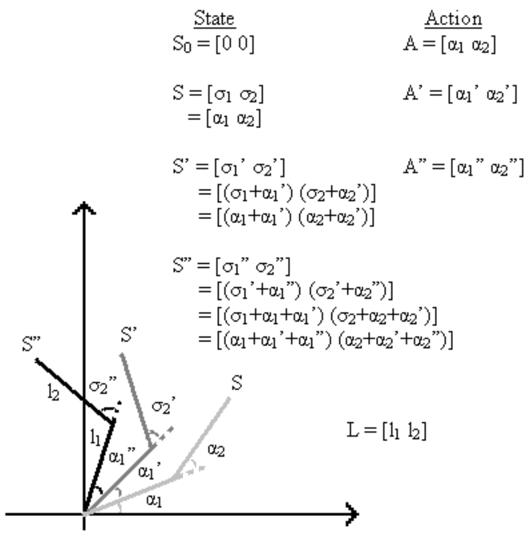

Figure 3. Example embodiment of a *Rabit* agent. A two-joint robotic arm, with arm segments of length l_1 and l_2, moving from state S_0 (arm completely outstretched along the horizontal axis) to state S to state S' to state S'', as it sequentially performs actions A, A', and A''. Note that the effects are not shown in this figure.

between the *previous state* and the *current state* (the state of the arm after the current action was executed). As a result of the possible actions, the absolute angle at each joint can be anywhere in the range of 0° to 360° (modulo 360°) but only in multiples of 10°.

The end tip of the arm leaves a trail of 'paint' on the workspace, as it moves along the workspace. The *effect* of an action is defined as a directed straight line segment connecting the end tip of the previous and the current states of the arm (approximating the paint trail). The effect can be considered as a vector of displacement $E=(x_c-x_p,\ y_c-y_p)$, where $(x_p,\ y_p)$ and $(x_c,\ y_c)$ are the end tip coordinates for the previous and current state respectively. In the figures described in the sections below, a behavior can be visualized by the trace of the robotic arm's end tip as it moves around the workspace.

Figure 4 shows traces for four different behaviors of a 3-joint robotic arm. Each model's behavior consists of 72 actions, and starts with the arm fully outstrechted horizontally to the right and ends in the same position. Figure 4a shows the trace of a model that repeatedly (thirty-six times) performs the action sequence [10,0,0] followed by [0,10,0], or, notationally $([10,0,0]\ [0,10,0])^{36}$, returning the arm to its starting position. The model in 4b's behavior is $([10,10,0]^9\ [0,10,0]^9)^4$, consisting of four repetitions of an 18-step action sequence, which in turn consists of nine repetitions each of two component actions. Similarly, the behaviors in 4c and 4d are respectively given by the action sequences $([0,10,0]^9\ [10,-10,10]^9)^4$ and $([10,0,0]^9\ [0,0,10]^9)^4$.

6.2 How versus What to Imitate

Separate metrics can be used for measuring the similarity of states, actions, and effects. (See Appendix A for formal definitions of the μ^{action}, μ^{state}, and μ^{effect} metrics employed for gauging similarity in each of these aspects.)

These metrics can also be combined in weighted sums to yield more complex metrics taking more than one of these aspects into account. The choice of the metric determines, in part, *what* will be imitated, whereas solving the correspondence problem concerns *how* to imitate (Dautenhahn & Nehaniv 2002). In general, aspects of state, action and effect as well as level of granularity of matching could all play roles in the choice of metric for solving the problem of how to imitate (Nehaniv & Dautenhahn 2001, Alissandrakis, Nehaniv & Dautenhahn 2002, Nehaniv 2003b), and the metrics used were chosen to measure these aspects (at fine granularity) in our test-bed. A discussion of

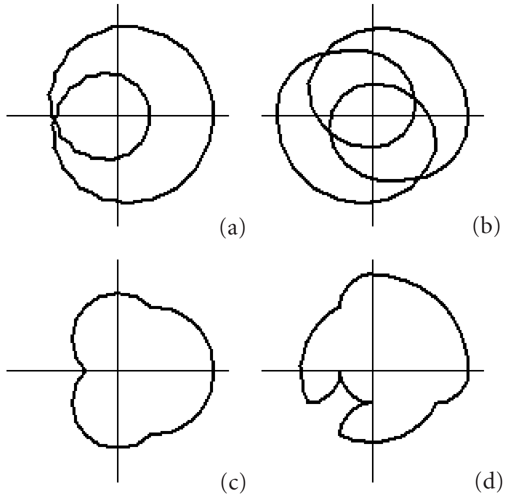

Figure 4. Traces of four different examples of model behaviors. Shown are the effect trails created by the end tip of the model agent manipulator arm after executing a complete behavioral pattern. All model agents shown have the same embodiment $L = [20,20,20]$. These behaviors consist of sequences 72 actions in length that return the arm back to its initial position. See text for detailed descriptions of the action sequences comprising these behaviors. Pattern (c) is used for the experiments reported in this paper on proprioception, loose perceptual matching, and synchronization, while patterns (c) and (d) are used for the experiments on imitators which change their embodiment during learning. The experiments and results are presented in Section 7 below.

granularity and its relationship to the action space available to imitators is given in Appendix B.

The robotic-arm agents in *Rabit* each attempt to solve a correspondence problem using a given metric, but do not generate or adapt their own metrics in the experiments reported here. On-going research in robot imitation is addressing the complementary problem of how to extract agent subgoals and generate suitable metrics automatically (Nehaniv & Dautenhahn 2001, Nehaniv 2003b, Billard & Schaal 2001). In the present work, subgoals are tacitly given by providing appropriate perceptual keys for the imitator to match, one for each action of the model.

6.3 Growth of the correspondence library

The model's behavioral pattern may naturally be broken down as a sequence of actions that each move the robotic arm of the agent from the previous state to the current state, while leaving behind a trail of paint as the effect. The nature of the experimental test-bed, with the fixed-base rotary robotic arms, favors cyclical looping effects and the model patterns used in the experiments were designed to be of this form (Figure 4). Each complete behavioral pattern that returns the model's arm to its initial state observed by the imitator is called an *exposure*, and the imitator is exposed to repeated instances of the same behavioral pattern. At the beginning of each new exposure it is possible to reset the imitating agent to the initial state. This resetting is called *synchronization* in our experiments. Note that this definition of synchronization is a great simplification if compared to synchronization of behavior among humans or other animals (cf. Section 4 above).

The relationship among the components of ALICE is shown in Figure 1 above. The correspondence library is initially empty. At each time step, i.e. for each action of the model, the imitator agent may be able to perceive the model's action, previous and current state and also the effect. The agent might perceive any of those aspects or a combination depending on the metric it is using.

The first time a percept occurs, a new entry is created in the correspondence library with that percept as its indexing key. When created, the key for the entry contains the data on the perceived subset of the observed model's action/state/effect and/or the state of the imitator, as perceptual and proprioceptual components respectively. (Which perceptual components are used in the keys depends on the metric.)

A randomly generated action is initially used as the corresponding action the first time the perceptual key is encountered and is stored under that key. In *Rabit*, as the agents are robotic arms, a natural choice for a generating mechanism would be some kind of inverse kinematics algorithm. But since the ALICE framework can use *any* generating mechanism as long as it can explore the entire search space, a generating mechanism that returns *random* (but valid) actions is used instead as this allows us to more clearly evaluate the contribution of ALICE, rather than a sophisticated generating mechanism, to imitative performance. For a given agent embodiment $L=[l_1, l_2, l_3\ldots, l_n]$, this generating mechanism will return a single action $A=[\alpha_1, \alpha_2, \alpha_3, \ldots, \alpha_n]$, where every α_i will be chosen randomly from between $\{0°, -10°, +10°\}$. Over time, several actions can be associated with the same perceptual key in the library.

The percepts an imitator experiences will determine the number of keys in the library, and thus the number of entries and the size of the correspondence library that is being built-up while the imitator tries to imitate the model. For example, if the key only contains actions, and the entire model behavior is composed of ten different actions, then only ten entries will be created in the library. But if instead the keys also (or *only*) contained states, then the number of entries will be equal to the number of different states the model agent goes through during the performance of its behavior, as the number of states will be greater (or equal) to the number of actions.

The number of library entries will directly affect the learning rate, especially since the generating mechanism on its own is highly unlikely to find proper corresponding solutions. At any stage, in the development of the correspondence library, the entries need initially to be scanned to discover if the currently perceived model information is related to an entry key and then that entry is updated or created. The more entries per model behavior, the slower they will be updated and more exposures to a complete model behavior will be required in order for them to contain appropriate correspondences.

Each key consists of exteroceptive (the model's action, state and effects) and/or proprioceptive (the imitator's own state) entry fields. When the model's action triggers an existing perceptual key,[4] e.g. if it has been observed before, then there will also be at least one corresponding action in a correspondence library entry. Using the metric, the predicted results of actions proposed by the generating mechanism (random in this implementation) are compared with the predicted results of ones from the library associated with the perceptual key, and a best one from among this set of proposed actions is executed. If this executed action was the newly generated one, it is added to the correspondence library entry.

The actions stored in the library with particular keys are thus used as partial solutions to the correspondence problem. New actions proposed by the generating mechanism at each time step might enter the correspondence library as described above. It is possible to employ a more complex action proposal mechanism (e.g. inverse kinematics) than a random generating mechanism, and, indeed, Alice is designed to accommodate any generating mechanism that returns valid actions from the search space.[5]

Two ways to speed up the learning rate are either (1) to limit the number of entry key fields, or (2) to use a *similarity threshold* when comparing the currently perceived model behavior aspects and the keys in the correspondence library. With the latter, more than one percept combination will match a given key, and therefore candidate matching actions in the library will be updated more frequently. The resulting performance will depend on the fidelity of these aspects in the model behavior. For example if the agent's movements are concentrated only in a small portion of the workspace, high fidelity is required to capture the details. If, on the other hand, the agent makes broad movements, locally accuracy is not that important.

Controlled by a threshold, it is thus possible not to require an *exact match* for the perceptual and/or the proprioceptive components of the trigger key, but a loose one that is 'close enough' according to the metric. We call this *loose perceptual matching*, and we hypothesized that it should support learning and generalization. Note that our simulations use a simple implementation of the 'close enough' judgement which, in humans and other animals could be based on environmental or context dependent factors, as well as motivational, cognitive and various other factors (e.g. urgency to find a matching behavior, personal goals of how precisely one should imitate, etc.).

In order to speed up the learning, it is possible to generate more than one action per time step and choose a best one (according to a metric used — see above).[6]

7. Experiments and Results

Experiments were performed to evaluate the contribution to the learning rate (using the ALICE framework) of *loose perceptual matching* (described in Section 7.1), *synchronization* (described in Section 7.2) and *proprioceptive matching* (described in Section 7.3) in the early stages of learning.

All the agents had the same embodiment ($L=[20,20,20]$). The metric used by both imitators was $\mu_{error} = ½(\mu^{action}+\mu^{state})$, which combines an evaluation of both action and state matching (see Appendix A). Results for other behaviors, metrics, and robotic-arm embodiments were similar to those in the experiments described here.

7.1 Loose perceptual matching

It was hypothesized that using *loose* perceptual matching for entries in the correspondence library (depending on a *similarity threshold*) instead of using *exact* perceptual matching, would result in more generic entries that would improve the learning rate.

When an imitator agent looks up a perceptual key in its correspondence library to find the relevant entry to the currently perceived model's action, state and effect and the imitator's own state, it can require an exact match, or one that is close enough (a loose match that depends on a similarity threshold). A series of experiments was conducted to examine the effect that using loose perceptual matching can have on the learning rate during the early stages of learning. Ten experimental runs were conducted under the same conditions. Each run consisted of ten exposures to a model's behavior for two imitating agents trying to imitate a model agent, one of them requiring an exact match for the keys in the correspondence library and the other one accepting a 10% margin of looseness. Both imitators synchronized with the model and did not use proprioceptive matching.[7]

The values of this error metric for the imitator using loose perceptual matching are plotted in Figure 5 (top panel), and for the imitator using exact matching in Figure 5 (middle panel). The top panel shows the value of the error metric used reducing over time, while the middle panel shows the value of the metric remaining relatively constant during the same initial learning stages. This supports our hypothesis that loose perceptual matching can improve the learning rate at the early stages. The ratio of the average error of the imitator that uses loose matching over the average error of the imitating agent that requires an exact match can be seen in the bottom panel of Figure 5 constantly decreasing and below 1, showing that the numerator (average error with loose matching) is minimized faster than the denominator (average error with exact matching) and indicating a comparative many-fold reduction of error when loose perceptual matching is used.

This is explained by the fact that there are fewer, and more generic, entries in the correspondence library of the imitator using the loose matching, resulting in a faster improvement of performance. When exact matching is used, a significantly longer learning period is required.

7.2 Synchronization

Inspired by the biological and psychological importance of synchronization (cf. Section 4), we hypothesize that it might lead to improved techniques in solving the correspondence problem. Synchronization was implemented in the *Rabit* test-bed as follows: at the end of each successive exposure of the imitating agent to the model, the imitator arm can be reset to the same initial position, thus

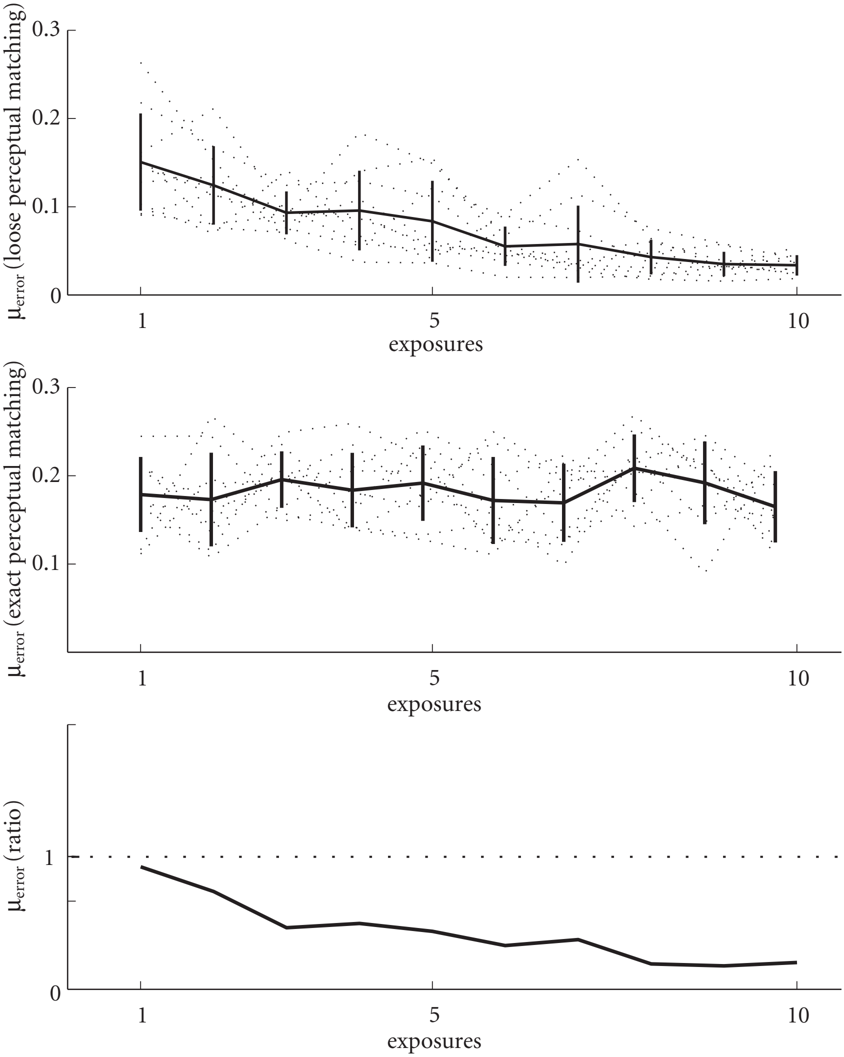

Figure 5. Loose perceptual matching experiments. The μ_{error} value (thicker line shows average value for ten experimental runs, vertical bars indicate the standard deviation, dotted lines indicate values of individual runs) over ten exposures when using loose perceptual matching (top panel) and when using exact matching (middle panel). The bottom panel shows the ratio of the average μ_{error} using loose (top) over using exact (middle) perceptual matching.

synchronizing the imitator's behavior with the model. A series of experiments was conducted to assess its efficacy. Ten experimental runs were conducted under the same conditions. Each run consisted of ten exposures to a model's behavior for two imitating agents trying to imitate a model agent, one of them synchronizing after each exposure to match the initial state of the model, and the other not. Each run lasted ten exposures and both agents used a similarity threshold of 10% for perceptual matching and no proprioceptive matching.[8]

In Figure 6, the top panel shows the value of the error metric reducing very quickly over time, while the middle panel shows the value of the metric remaining constant during the same initial learning stages. The ratio of the average error of the imitator that synchronizes with the model over the average error of the imitating agent that does not can be seen in Figure 6 (bottom panel) constantly decreasing and below 1, indicating that the numerator (average error using synchronization) is minimized faster than the denominator (average error not using synchronization). This supports our hypothesis that if the imitator synchronizes with the model, this can benefit the imitative learning.

It is difficult for an imitator that does not synchronize to reach again states relevant to the model behavior if the initial imitation attempts are not successful. Soon after the first exposure to the model, the imitator that does not synchronize with the model becomes 'lost' due to cumulative errors that are not corrected by resetting (Figure 6, middle), while the imitator using synchronization shows steady improvement (Figure 6, top). The non-synchronizing imitator will require a far greater number of exposures to return (via a partly random walk using the generating mechanism) back 'on track' and from that point onwards successfully imitate.

The focus of these experiments was to investigate the importance of synchronization in the early stages of learning, up to the point that one of the imitators is able to achieve a satisfactory performance. Further studies are needed to investigate how non-synchronizing imitator agents might eventually improve imitative performance over much longer simulation runs.

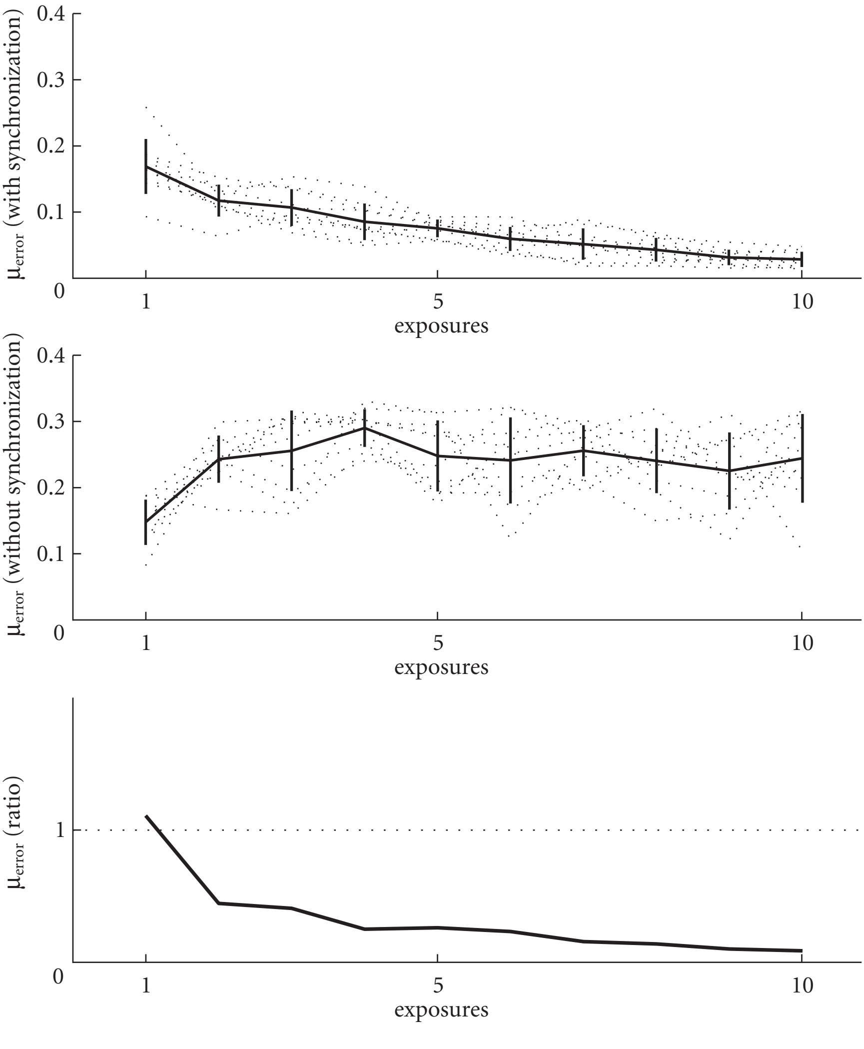

Figure 6. Synchronization experiments. The μ_{error} value (thicker line shows average value for ten experimental runs, vertical bars indicate the standard deviation, dotted lines indicate values of individual runs) over ten exposures when an imitator synchronizes with the model (top panel) and when another does not (middle panel). The bottom panel shows the ratio of the average μ_{error} using (top) over not using (middle) synchronization.

7.3 Proprioceptive matching

All species that imitate have very sophisticated proprioception, which might help in relating the model's states to their own states. Some proprioception is always used in the Alice framework whenever perceptual keys include a state or effect component, since the imitator's own state is taken into account when calculating the metric values for the different possible actions, but no proprioception is used if the correspondence library key consists solely of the action component. The correspondence library entry keys may contain both exteroceptive (the model's action, state and effects) and proprioceptive (the imitator's own state at the time of the observation) information. It was hypothesized that always using a proprioceptive component, giving the imitator's state at the time of the observation of the model in the entry key, would improve imitative performance.

Ten experimental runs were conducted under the same conditions. Each run consisted of ten exposures to a model's behavior for two imitating agents trying to imitate a model agent, one of them using proprioceptive matching, the other not. Each run lasted ten exposures, both agents were synchronized and used a similarity threshold of 10%.[9]

The top panel of Figure 7 shows the value of the metric used reducing over time, while the middle panel shows the value of the metric reducing over time faster during the same initial learning stages. This contradicts our hypothesis that proprioceptual matching can imporve the learning rate at the early stages of learning.

The ratio of the average error per exposure of the imitating agent that uses proprioceptive matching over the average error of the imitating agent that does not can be seen in Figure 7 (bottom panel). Contrary to the hypothesis, the ratio is increasing, and only towards the end of the experiments seems to be decreasing. This indicates that the numerator (average error using proprioceptive matching) is minimized slower than the denominator (average error not using proprioceptive matching).

Utilizing a proprioceptive matching component for the keys in the correspondence library increases the number of keys K, on which the search space depends, exponentially by a factor equal to the number of all possible states of the imitator (see Section 6.3 and Appendix B). Similar to the experimental results for loose matching described in Section 7.1, ignoring the proprioceptive matching for the keys reduces the number of entries in the library, thus allowing them to update more frequently, resulting in a faster improvement of

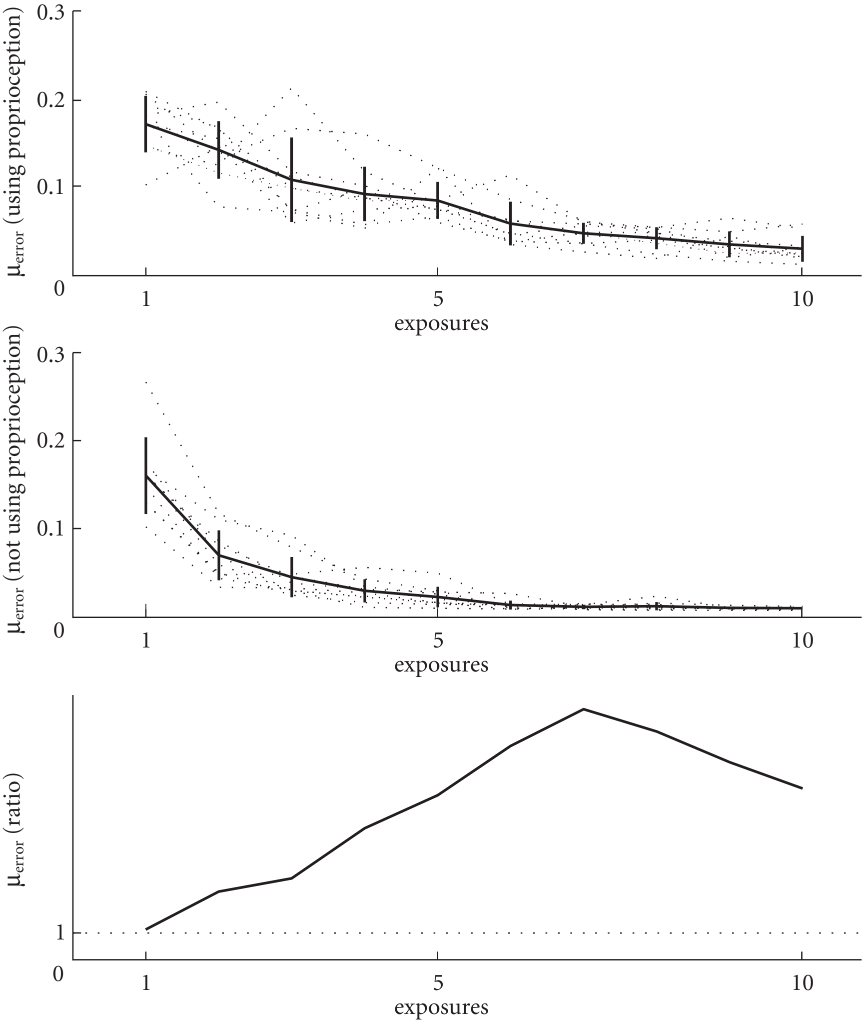

Figure 7. Proprioceptive matching experiments. The μ_{error} value (thicker line shows average value for ten experimental runs, vertical bars indicate the standard deviation, dotted lines indicate values of individual runs) over ten exposures when an imitator does not use proprioceptive matching (top panel) and when another does (middle panel). The bottom panel shows the ratio of the average μ_{error} using (top) over not using (middle) proprioceptive matching.

performance. Note, it is difficult to relate these results on proprioceptive matching directly to biological systems where proprioception might facilitate learning in a variety of ways, while in our experimental test-bed (in which computation is sequential) it necessarily increased the search space.

7.4 Embodiment modification during learning

The correspondence problem draws our attention to the fact that the model and imitator agents can have dissimilar embodiments with different affordances, complicating the discovery of appropriate mappings to achieve similar actions, states and effects. These dissimilarities might be static, due to the fact that the agents belong to different 'species', but can also be dynamic, caused by epigenetic factors like body development or accidents/modifications that can happen during the agent's lifetime and result either in restriction or augmentation of its capabilities.

The following two examples illustrate that the ALICE framework is reasonably robust to tolerate embodiment changes during the early learning stage, and can also adjust to embodiment changes during the later learning stages, if the imitator continues to be exposed to the model.

Figure 8 shows an imitator that is using the ALICE framework to learn how to imitate a model behavior while its body is 'growing'. Every two exposures to the model, the length of each imitator arm segment is increased by a single unit, while the model embodiment remains constant ($L_{\text{model}} = [20,20,20]$). Initially (upper left corner) the imitator is half the size of the model ($L_{\text{imitator}} = [10,10,10]$). After twenty exposures (lower right corner) the imitator has 'grown' to the same size as the model ($L_{\text{imitator}} = [20,20,20]$). Even though its embodiment is changing during the early learning stage, the imitator is able to associate corresponding actions, achieving a perfect imitation of the model behavior.

In this particular example, the imitation performance is partially helped by several factors. The first factor is that the model and the imitator agent embodiments have the same number of (equal-length) joints, and the segment length modifications were uniform. Also, the imitator is considering the *actions* aspect of the model behavior only, and this allows the library entries to be updated more frequently, as the imitator has to learn to correspond to only *two* different actions, independent of the context they appear in. The synchronization with the model guarantees that once these actions are learned, performing them in the right sequence (defined by the behavior) will result in the imitator achieving similar states and effects as the model.

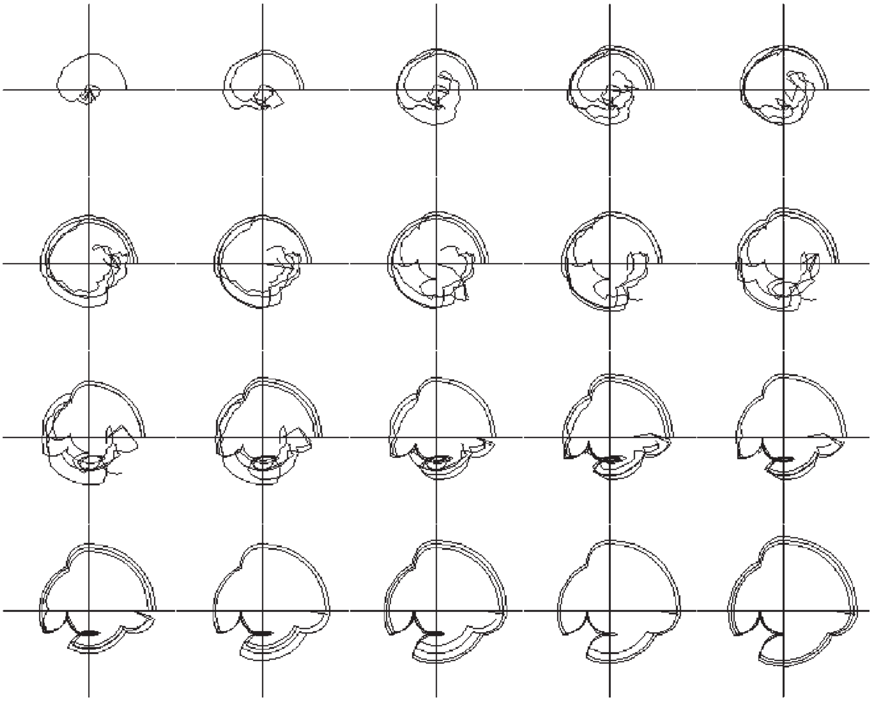

Figure 8. An example of learning during embodiment 'growth'. The trace of the imitator's behavior is shown (left to right, top to bottom, each screenshot taken after each exposure to the model, showing the current and the three previous patterns) when exposed to a model with a constant embodiment ($L_{\mathrm{model}} = [20,20,20]$) exhibiting behavior (d) from Figure 4. The imitator embodiment initially is $L_{\mathrm{imitator}} = [10,10,10]$ with each segment length increased by a single unit every two complete exposures. At the end of twenty exposures, imitator and model agents share the same embodiment. The imitator uses the *action* metric.

The *state* and *effects* aspects are more sensitive to embodiment changes (from a proprioceptive perspective). Their use is not necessarily restricting, but will demand a longer period between the embodiment changes, until the imitator has satisfactory learned the model actions with the current embodiment.

If the imitator agent continues to be exposed to the model, even after it has successfully learned the model behavior, the ALICE framework can be used for adapting the correspondences to embodiment modifications at that later learning stage.

Figure 9 shows an imitator that has already learned to imitate the model behavior. The imitator and the model agent had the same constant embodiment

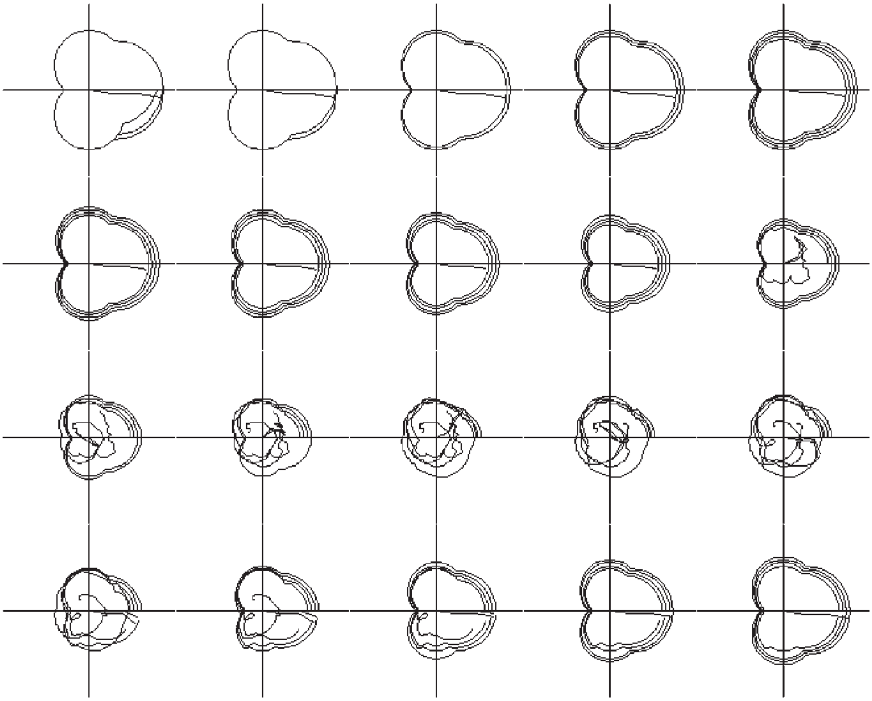

Figure 9. An example of adaptation to embodiment modification during learning. The behavioral trace of an imitator exposed to a model with a constant embodiment ($L_{\text{model}} = [20,20,20]$) exhibiting behavior (c) from Figure 4. The imitator embodiment initially is $L_{\text{imitator}} = [20,20,20]$ and has already at that stage (after twenty exposures) successfully learned the model behavior. Each segment length is decreased by a single unit after each exposure until its embodiment becomes $L_{\text{imitator}} = [10,10,10]$, and then is increased back to the original embodiment. The imitator uses the *action* metric.

during the early learning stage ($L = [20,20,20]$) for twenty exposures. At this point (upper left corner), the imitator starts to 'shrink' by a single length unit after every exposure until the embodiment is half the size of the model ($L_{\text{imitator}} = [10,10,10]$). It then starts to 'expand' until it achieves again its original size. Here, similar to the example shown in Figure 8, the imitator only considers the *action* aspect of the model behavior, allowing for a short period between embodiment changes (in this case, only a single exposure).

Note that at smaller sizes, the particular action metric (which was not normalized for size differences) does not distinguish well between actions of the small imitator and its larger model. But the imitator is able to recover when the embodiment starts to 'grow' again back to the original size.

7.5 Limitations of *Rabit*

The *Rabit* test-bed is more complex and closer to a real-world platform than *Chessworld*, but still has several limitations.

Two limitations *by design*, the restriction of the possible angle values for the agent actions, and the consideration of correspondences of *single* actions only (instead of correspondences of *sequences* of actions), result in the use of a single level of granularity. These decisions were taken in an attempt to restrict the search space, in order to speed up the learning rate without considering specific optimizations. Even with these limitations, it was possible to illustrate that the Alice framework is able to solve the correspondence problem for the dissimilar agents of the *Rabit* test-bed. A next version of the test-bed could be implemented, removing these restrictions and allowing to study the correspondence problem for the *Rabit* agents using different granularities as well.

Further limitations include the two-dimensional nature of the test-bed (although it could easily be extended to three dimensions at the cost of increasing the search space), and the limited types of 'meaningful' behavior available for imitation (not only sequences of actions that create looping patterns).

A more complex test-bed, possibly in *hardware*, could allow for interactions with objects in the environment and more realistic physics.

8. Cultural transmission among heterogeneous agents — The beginning of culture in robots?

Imitation broadly construed is required for cultural transmission (Dawkins 1976: Ch. 11). Transmission of behavioral skills by social learning mechanisms like imitation may also be fundamental in non-human cultures, e.g. in chimpanzees (Whiten, Goodall, McGrew, Nishida, Reynolds, Sugiyama, Tutin, Wrangham & Boesch 1999), or whales and dolphins (Rendell & Whitehead 2001). Note, every animal is an individual and no two individuals possess exactly the same embodiment. Two main issues that must be studied to assess potential for imitation to serve as a basis for culture in artificial systems are whether patterns can be transmitted among dissimilarly embodied artificial agents, and if so, whether variations occur during these replications. These are properties of *replication* and *variability* in behavior that are prerequisites for the evolution of culture (or its pre-cursors), cf. (Dawkins 1976, Bonner 1983, Nehaniv 2003a).

8.1 Behavioral replication and variability

The *Rabit* test-bed in the context of the ALICE framework can be used to study the social transmission of a behavior via imitation. An imitator of a model might in turn also be imitated by another agent, creating chains or networks of social transmission for an original model behavior pattern.

The example illustrated in Figure 10 demonstrates exactly such a transmission of a behavioral pattern via social learning in a chain of imitating agents. The original model *model0* with three joints is shown in Figure 10, left. It is imitated by a two-joint robotic arm *imitator0* (Figure 10, centre), which in turn is imitated by another imitator *imitator1* (Figure 10, right) with the same embodiment as the original model, but which only perceives the behavior of *imitator0*. Both imitators' metrics only consider the *action* aspect of the behavior of their respective models.

After transmission through the intermediary, the behavioral pattern that has been acquired by the second imitator in Figure 10 (centre) is quite similar to the original despite differences in embodiment in the chain of transmission. However, variation has been introduced even between the behavior as exhibited by the model and the second imitator (which have the same embodiment). The second imitator is using different actions (as compared to those of the original model) to achieve the behavioral pattern.

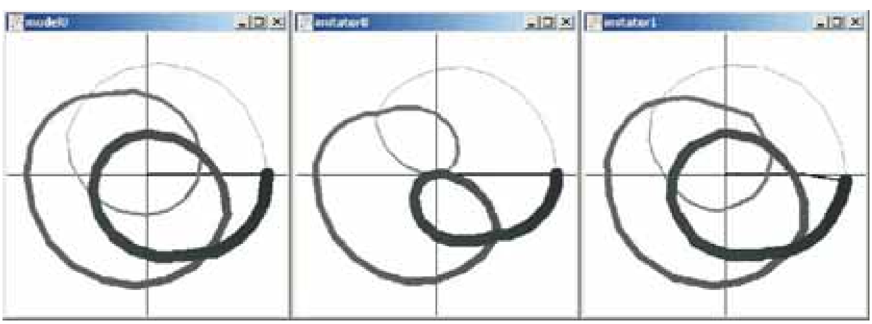

Figure 10. An example of cultural transmission with variation among heterogeneous agents. The original 3-DOF model is *model0* (left, $L=[20,20,20]$), with 2-DOF *imitator0* (middle, $L=[30,30]$) acting in turn as a model for 3-DOF *imitator1* (right, $L=[20,20,20]$). The model exhibits behavior (b) of Figure 4 which consists of 72 actions, but only two different ones. Both imitators use the *action* metric, and eventually are using only two actions to match the two different actions of they observe. At the end of the chain the behavior appears qualitatively very similar to the original model's, but is actually accomplished using different actions than those of the original model.

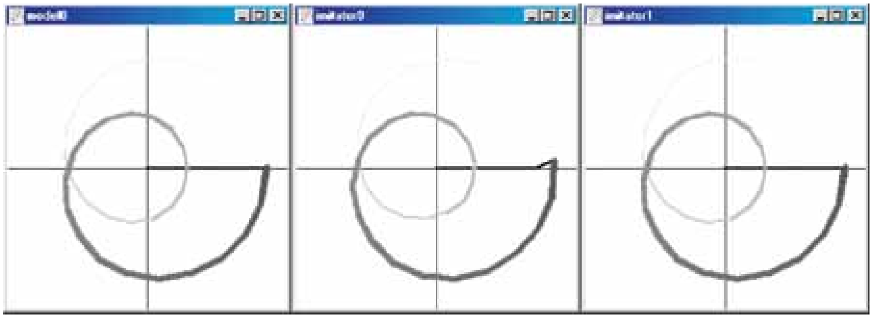

Figure 11. An example of perfect cultural transmission among heterogeneous agents. The original 3-DOF model is *model0* (left, $L=[20,20,20]$), with a 6-DOF *imitator0* (middle, $L=[10,10,10,10,10,10]$) acting in turn as a model for 3-DOF *imitator1* (right, $L=[20,20,20]$). Both imitators use the *action* metric. The behavior imitated is (a) of Figure 4. Eventually the imitators are each using only two actions from their correspondence library to match the two distinct actions they each observe comprising their respective model's 72 action behavior. The behavior is successfully transmitted along the chain with no variation when it has passed through the chain. Arriving at the third robotic-arm the behavior has been replicated faithfully and is performed exactly as by the original model.

This first example shows that transmission with variation of a behavioral pattern is possible through a chain of robotic agents, despite differences in embodiment of agents involved. This and the other simple examples here serve as proof of the concept that by using social learning and imitation, rudimentary cultural transmission with variability is possible among robots, even heterogeneous ones.

Similarly, Figure 11 shows transmission of a behavior from a model with 3-joints to an imitator with 6-joints which then transmits it to another imitator with 3-joints. In this case, the 6-joint imitator eventually learns to 'freeze' alternating joints and uses only three of its six degrees of freedom to match the behavior, which is then reproduced by the 3-joint imitator exactly as it was carried out by original model. Note the slight bend at the end of the behavior of the 6-DOF imitator in the center at this stage, due to a non-zero angle in the last joint which has little effect on the action metric.

In contrast, Figure 12 shows a behavior that is transmitted from a 6-joint model to a 3-joint imitator and then to another 6-joint imitator. The 3-joint imitator, with fewer degress of freedom, is unable to match the pattern faithfully, but its behavior then appears to be matched quite faithfully by its 6-joint

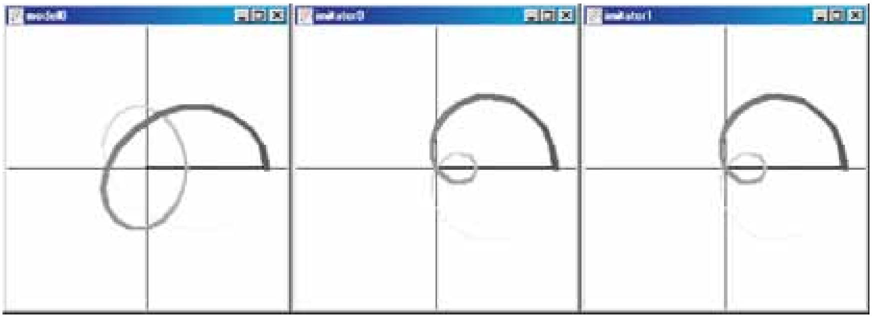

Figure 12. An example of cultural transmission among heterogeneous agents with simplification of transmitted behavior. The original 6-DOF model is *model0* (left), with a 3-DOF *imitator0* (middle) acting in turn as a model for its 6-DOF *imitator1* (right). Both imitators use the *action* metric. The behavior is succesfully transmitted through the chain, but *imitator0*, with fewer degrees of freedom than the original model, has 'simplified' the behavior, which its 6-DOF imitator (*imitator1*) faithfully then reproduces. The 6-DOF model's behavior is given as $[10,-10,0,-10,-10,0]^{36}$, a repeated single action behavior. Here *imitator1* has learned to only use three of its six joints and keep the other three fixed. Eventually the two imitators are only using a single entry from their correspondence libraries to match their respective model's single repeated action.

imitator. In this case of the behavior passing through a less complex imitator, it becomes 'simplied' in the course of transmission.

8.2 Emergence of proto-culture

Examples illustrated in Figures 13, 14, 15 and 16 demonstrate the emergence of 'proto-culture' in a cyclically ordered chain of six (Figures 13 and 14) or three (Figures 15 and 16) imitators with *no overall model.* The agents imitate only the agent counter-clockwise from them, using the *action* metric. Initially they move randomly, as the generating mechanism is trying to discover correspondences for the (also random) actions of their model. Over time, they are able to imitate each other's actions and a stable behavioral pattern emerges.

Different runs yield different emergent culturally sustained behaviors. The location and orientation of the emergent pattern is different in each agent's workspace, since the location and orientation are irrelevant to the action metric; they will depend on the state of the agent at the moment that it has solved its correspondence problem. Each agent's state will vary as a result of the agents not synchronizing.

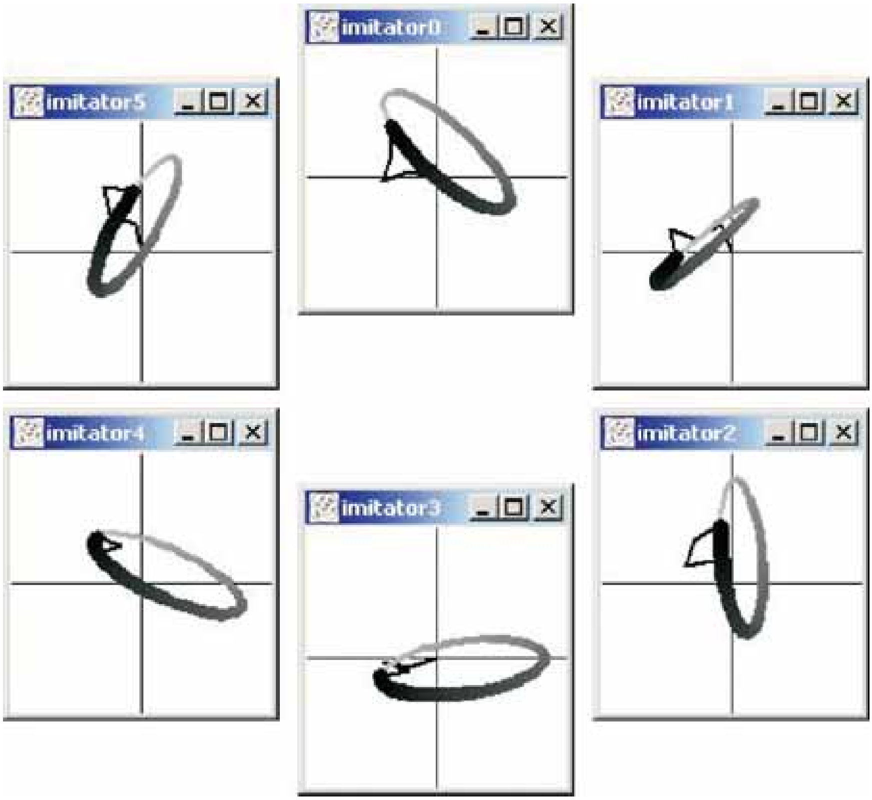

Figure 13. An example of a proto-culture emerging via imitation. A stable behavior pattern emerges as a result of each agent imitating the agent counter-clockwise from them, without an overall model or any particular behavior to imitate. All agents have the same embodiment with four segments ($L=[15,15,15,15]$).

Also, it should be emphasized that the choice of metric (the action metric in the examples here) can greatly affect both the speed with which a shared behavior emerges, as well as its character. Using a metric dominated by its action component leads much more quickly to convergence. Metrics dominated by effects most readily converge to taking no action (since for all robotic arm agents, a 'zero effect' is easy to achieve by simply keeping all joints fixed, i.e. by the trivial action that changes each joint angle by 0 degrees). With state metrics, convergence will always take much longer than for the action metric, due to the large spaces of possible states for each robotic arm. In the proto-culture emergence experiments here, the robotic arms settle into repeating a single action. This is due to the fact

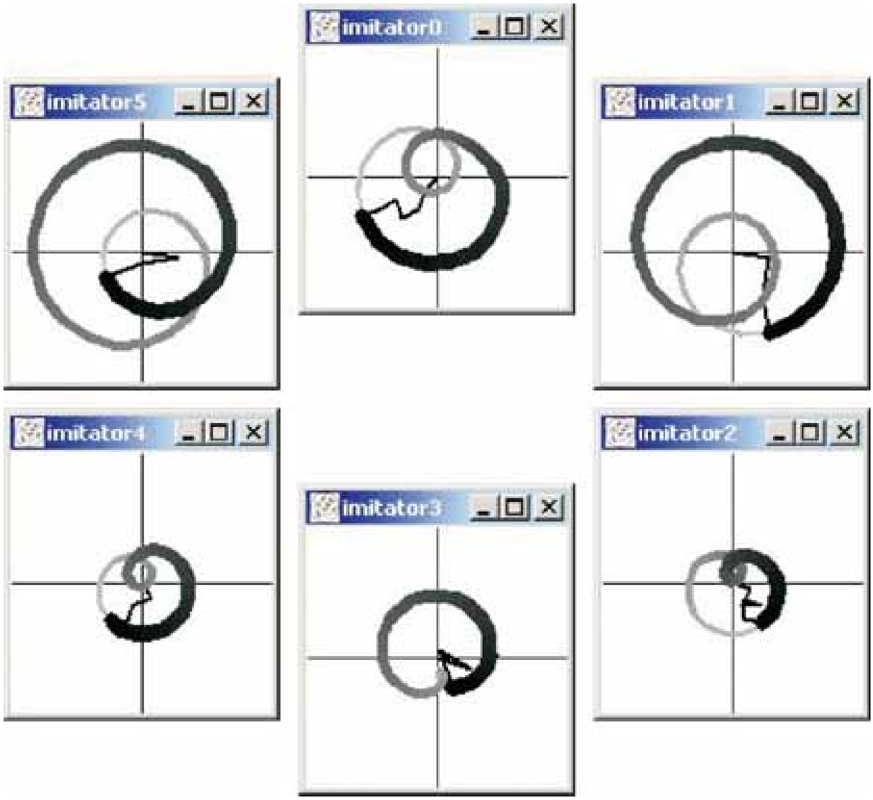

Figure 14. An example of an emerging proto-culture via imitation. A stable behavior pattern emerges as a result of each agent imitating the agent counter-clockwise from them, without an overall model or any particular behavior to imitate. The agents have dissimilar embodiments, imitators 0, 2 and 4 have embodiments $L=[20,20,20]$ with three segments, imitators 1, 3 and 5 have embodiments $L=[10,10,10,10,10]$ with five segments.

that the percepts — and, in this implementation, hence also the subgoals — to be matched are presented to the imitators at a single-action level of granularity and are matched with single actions rather than action sequences. (This is in contrast to the cultural transmission experiments in which the behavior of the model may be arbitrarily complex; although single-action-for-single-action correspondences are also used, the imitators attempt match an arbitrary sequence of various actions as long as the model persists in displaying it.)

More experiments would be needed in order to shed light on the detailed

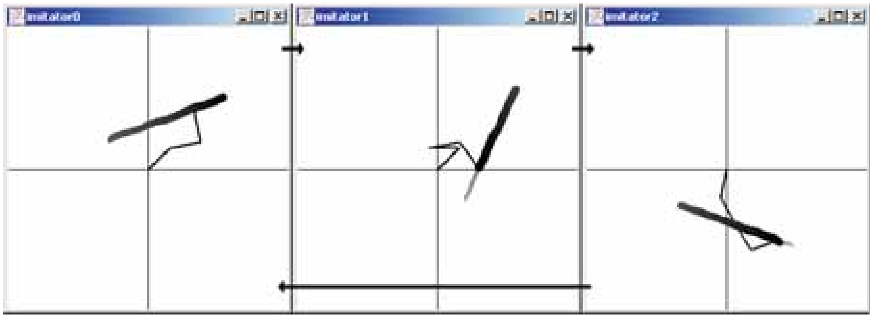

Figure 15. An example of an emerging proto-culture via imitation. A stable behavior pattern emerges as a result of each agent imitating the agent counter-clockwise from them, without an overall model or any particular behavior to imitate. All agents have the same embodiment ($L=[15,15,15,15]$, i.e. four segments). Arrows indicating the flow of behavioral transmission point from a model to its imitator.

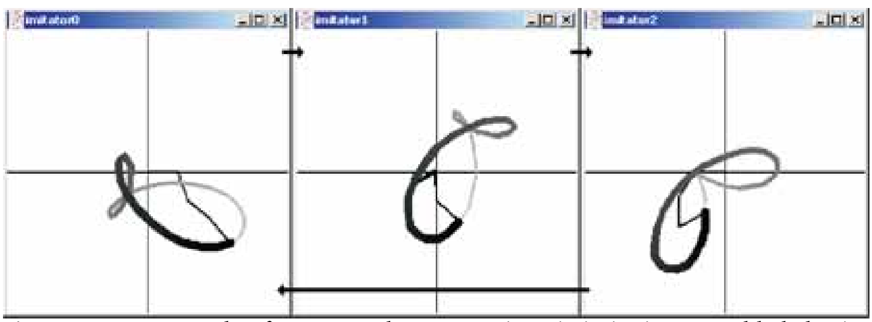

Figure 16. An example of a proto-culture emerging via imitation. A stable behavior pattern emerges as a result of each agent imitating the agent counter-clockwise from them, without an overall model or any particular behavior to imitate. All agents have the same embodiment ($L=[15,15,15,15]$ as shown in Figure 15). Arrows indicating the flow of behavioral transmission point from a model to its imitator.

nature and constraints on the emergence of proto-culture in our robotic arm test-bed and related systems.

9. Conclusions and Outlook

The experimental results of our experiments using Alice in differently embodied robotic arm agents suggest that (1) loose perceptual matching increases the rate of solving the correspondence problem ('how to imitate') significantly, (2) synchronization dramatically increases the rate of solving the correspondence problem, and (3) utilizing proprioreceptive matching for keys does not, at least for early stages of learning, increase the rate of the solution of this problem within our experiments (although it certainly does not prevent its solution). We also gave examples showing that ALICE can cope with dynamically changing embodiments in cases where the imitator shrunk or grew during the process of solving the correspondence problem, which has possible applications to fault-tolerance and self-repair by imitating agents.

Finally, we showed that ALICE allows the cultural transmission of behavioral patterns in a heterogeneous community of simulated robots, where diverse behavioral patterns and variations thereof might emerge. The cultural transmission of skills through a heterogeneous population of robots using the ALICE framework could potentially be applied to the acquisition and transmission of skills in more complex populations of robots, involved in carrying out useful tasks, e.g. on the shop-floor of a factory, with new robots coming and going, acquiring behaviors by observation without having to be explicitly programmed and without humans having to develop different control programs for different types of robots that need to perform the same task. Instead, the robots would autonomously create their own programs (using social learning) and correspondence libraries, even as new types of robots with different embodiments join or leave from the population.

In conjunction with our previous work using a chessworld test-bed (Alissandrakis, Nehaniv & Dautenhahn 2002), our results serve to establish the generalizability of the ALICE framework. Scalability in different settings depends on particularities of the embodiments, and the sophistication of the generating mechanism used (here, only random actions were needed) to propose candidate matching actions or action sequences, and also on processing speed and optimization issues for the specific platforms.

In technical terms, our future work in solving the correspondence problem will involve different robotic test-beds, as well as developing new methods for subgoal extraction and the automatic generation of metrics. It would be appropriate also to compare success of imitation at different threshold levels for loose perceptual matching, and to allow these thresholds to adapt to be loose enough for effective learning but not so loose as to compromise success. More generally, we wish to contribute to our understanding of skill-learning in robots, as well as establishing 'a step towards releasing robots from social isolation' (Dautenhahn 1994), i.e. equipping robots with necessary developmental and evolutionary prerequisites in order to become socially intelligent. Possible applications for such *social robots* are abundant, ranging from healthcare and entertainment to service robotics (cf. reviews in (Fong, Nourbakhsh & Dautenhahn 2003, Dautenhahn 2003)).

Simple (proto-) cultural societies whose members include artificial or robotic agents can harness social learning and imitation as *replication* mechanisms, while differences in embodiment can result in *variability*, thus providing a Darwinian substrate for an cultural evolvability (Nehaniv 2003a). This is illustrated here in quite minimal examples showing transmission and variation of behaviors, as well as the emergence of shared behaviors in a community of artificial robotic agents. More sophisticated applications of these principles to achieve greater evolvability properties, together with systematic artificial intelligence methods such as Alice for solving the correspondence problem (and other problems of social learning, cf. (Dautenhahn & Nehaniv 2002)), are natural directions to pursue next. It seems that we may indeed be heading toward cultures in robots, possibly including cultures in mixed societies involving robots, software agents, and biological agents such as humans.

Notes

1. Novelty can be achieved in our test-beds, e.g. by simply exposing a naive imitator (with Alice) that has never performed *any* action to a novel behavior of the model designed by the experimenter. Any matching action sequences performed by the imitator will then be novel. Similar remarks also apply to imitators exposed to model behaviors which consist of novel sequences of previously learned primitive or compound actions, as such imitators will automatically exhibit novelty of sequencing.

2. See also Nehaniv & Dautenhahn (1998, 2000, 2001) for a formal statement of the correspondence problem relating to the use of different error metrics, and for other applications.

3. *Rabit* was implemented using the *Swarm simulation system* (http://www.swarm.org). First results from this test-bed were presented in Alissandrakis, Nehaniv & Dautenhahn (2003a,b).

4. See Appendix B for consideration of how the number of keys and possible actions affect the size of the problem space in solving the correspondence problem.

5. The simple random generating mechanism performs well enough for test-bed purposes, although the rate of learning is naturally slower than for more complex action generation mechanisms. Sophisticated applications of Alice can benefit by replacing, in a modular way, this action generation with a more sophisticated one appropriate to the given application.

6. Not implemented in the current test-bed, but another possible part of Alice, is the *history mechanism*, which also considers sequences of past imitative attempts when updating the correspondence library entries, as previously used in the *Chessworld* test-bed (Alissandrakis, Nehaniv & Dautenhahn 2002).

7. The efficacy of this choice for synchronization and not using proprioceptive matching was seen in numerous pre-experimental runs and is illustrated by the experimental results in Sections 7.2 and 7.3.

8. The choice for loose matching and not using proprioceptive matching was made based on numerous preliminary pre-experimental runs. The efficacy of this choice is illustrated by the experimental results in Sections 7.1 and 7.3.

9. The choice for using loose matching and synchronization is explained by the experimental results from numerous pre-experimental runs, and their efficacy is illustrated by the experiments Sections 7.1 and 7.2.

10. This is effectively equivalent to assuming both arms are outstretched, and compares the result the actions have on the respective agents, ignoring their current states.

11. For example, for five joints, this gives 243 possible actions compared to 60466176 possible actions if c_i can take any value from between $-180°$ to $+180°$ in multiples of $10°$.

References

Alissandrakis, A. (2003), Imitation and Solving the Correspondence Problem for Dissimilar Embodiments — A Generic Framework, PhD thesis, University of Hertfordshire.

Alissandrakis, A., Nehaniv, C. L. & Dautenhahn, K. (2001), Through the Looking-Glass with ALICE — Trying to Imitate using Correspondences, in 'Proc. First International Workshop on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems, September 17–18, 2001, Lund, Sweden', Vol. 85, Lund University Cognitive Studies, pp. 115–122.

Alissandrakis, A., Nehaniv, C. L. & Dautenhahn, K. (2002), 'Imitation with ALICE: Learning to Imitate Corresponding Actions across Dissimilar Embodiments', *IEEE Trans. Systems, Man & Cybernetics: Part A* **32**(4), 482–496.

Alissandrakis, A., Nehaniv, C.L. & Dautenhahn, K. (2003a), Solving the Correspondence Problem Between Dissimilarly Embodied Robotic Arms Using the ALICE Imitation Mechanism, in 'Proc. Second International Symposium on Imitation in Animals and Artifacts — Aberystwyth, Wales, 7–11 April 2003', Society for the Study of Artificial Intelligence and Simulation of Behaviour, pp. 79–92.

Alissandrakis, A., Nehaniv, C.L. & Dautenhahn, K. (2003b), Synchrony and Perception in Robotic Imitation across Embodiments, in 'Proc.IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '03)', pp. 923–930.

Bentivegna, D. C. & Atkeson, C. G. (2002), Learning How to Behave from Observing Others, in 'Proc.SAB '02-Workshop on Motor Control in Humans and Robots: On the interplay of real brains and artificial devices, Edinburgh, UK, August, 2002'.

Billard, A. (1998), 'DRAMA, a connectionist model for robot learning: Experiments on grounding communication through imitation in autonomous robots', PhD Thesis, Depart. of Artificial Intelligence, University of Edinburgh.

Billard, A. & Dautenhahn, K. (1998), 'Grounding Communication in Autonomous Robots: an Experimental Study', *Robotics and Autonomous Systems* **24**(1–2), 71–81.

Billard, A. & Schaal, S. (2001), Robust learning of arm trajectories through human demonstration, in 'Proc. 2001 IEEE/RSJ International Conference on Intelligent Robots amd Systems (IROS 2001), Maui, Hawaii, USA, Oct.29–Nov.03 2001', pp. 734–739.

Billard, A., Hayes, G. & Dautenhahn, K. (1999), Imitation skills as a means to enhance learning of a synthetic proto-language in an autonomous robot, in 'Proc. AISB Symp. Imitation Animals Artifacts', Edinburgh, U.K., pp. 88–95.

Bonner, J.T. (1983), *The Evolution of Culture in Animals*, Princeton.

Breazeal, C. & Scassellati, B. (2002), 'Robots that imitate humans', *Trends in Cognitive Science* **6**, 481–487.

Butterworth, G. (1999), Neonatal imitation: existence, mechanisms and motives, in J. Nadel & G. Butterworth, eds, 'Imitation in Infancy', Cambridge University Press, pp. 63–88.

Byrne, R.W. (1999), 'Imitation without intentionality. Using string parsing to copy the organization of behaviour', *Animal Cognition* **2**, 63–72.

Dautenhahn, K. (1994), Trying to imitate — a step towards releasing robots from social isolation, in P. Gaussier & J.-D. Nicoud, eds, 'Proc. From Perception to Action Conference, Lausanne, Switzerland', IEEE Computer Society Press, pp. 290–301.

Dautenhahn, K. (1995), 'Getting to know each other — artificial social intelligence for autonomous robots', *Robotics and Autonomous Systems* **16**, 333–356.

Dautenhahn, K. (2003), 'Roles and Functions of Robots in Human Society — Implications from Research in Autism Therapy', *Robotica* **21**, 443–452.

Dautenhahn, K. & Nehaniv, C.L. (2002), The agent-based perspective on imitation, in K. Dautenhahn & C.L. Nehaniv, eds, 'Imitation in Animals and Artifacts', MIT Press, pp. 1–40.

Dautenhahn, K., Nehaniv, C.L. & Alissandrakis, A. (2003), Learning by Experience from Others — Social Learning and Imitation in Animals and Robots, in R. Kühn, R. Menzel, W. Menzel, U. Ratsch, M.M. Richter & I.O. Stamatescu, eds, 'Adaptivity and Learning: An Interdisciplinary Debate', Springer Verlag, pp. 217–421.

Dawkins, R. (1976), *The Selfish Gene*, Oxford.

Demiris, J. (1999), 'Movement imitation mechanisms in robots and humans', PhD Thesis, Depart. of Artificial Intelligence, University of Edinburgh.

Demiris, J. & Hayes, G. (1996), Imitative Learning Mechanisms in Robots and Humans, in 'Proc. of the 5th European Workshop on Learning Robots, Bari, Italy, July 1996', pp. 9–16.

Demiris, J. & Hayes, G. (2002), Imitation as a Dual-Route Process Featuring Predictive and Learning Components: A Biologically-Plausible Computational Model, in K. Dautenhahn & C.L. Nehaniv, eds, 'Imitation in Animals and Artifacts', MIT Press, pp. 327–361.

Fong, T., Nourbakhsh, I.R. & Dautenhahn, K. (2003), 'A survey of socially interactive robots', *Robotics and Autonomous Systems* **42**(3–4), 143–166.

Fragaszy, D.M. & Visalberghi, E. (1990), 'Social processes affecting the appearance of innovative behaviors in capuchin monkeys', *Folia Primatologica* **54**, 155–165.

Gallese, V., Fadiga, L., Fogassi, L. & Rizzolatti, G. (1996), 'Action recognition in the premotor cortex', *Brain* **119**, 593–609.

Gaussier, P., Moga, S., Banquet, J.P. & Quoy, M. (1998), 'From Perception-Action Loops to Imitation Processes: A Bottom-up Approach of Learning by Imitation', *Applied Artificial Intelligence Journal, Special Issue on Socially Intelligent Agents* **12**(7–8), 701–729.

Hayes, G. & Demiris, J. (1994), A robot controller using learning by imitation, in 'Proc. International Symposium on Intelligent Robotic Systems, Grenoble', pp. 198–204.

Herman, L.M. (2002), Vocal, social, and self imitation by bottlenosed dolphins, in K. Dautenhahn & C.L. Nehaniv, eds, 'Imitation in Animals and Artifacts', MIT Press, pp. 63–108.

Heyes, C.M. (2002), Transformational and Associative Theories of Imitation, in K. Dautenhahn & C.L. Nehaniv, eds, 'Imitation in Animals and Artifacts', MIT Press, pp. 501–523.

Heyes, C.M. & Ray, E.D. (2000), 'What is the signtificance of imitation in animals?', *Advances in the Study of Behavior* **29**, 215–245.

Kuniyoshi, Y., Inaba, M. & Inoue, H. (1994), 'Learning by watching: Extracting reusable task knowledge from visual observations of human performance', *IEEE Trans. Robot. Automat.* **10**, 799–822.

Kuniyoshi, Y., Inoue, H. & Inaba, M. (1990), Design and implementation of a system that generates assembly programs from visual recognition of human action sequences, in 'Proc. IEEE International Workshop on Intelligent Robots and Systems IROS '90', pp. 567–574.

Matarić, M.J., Jenkins, O.C., Fod, A. & Zordan, V. (2000), Control and Imitation in Humanoids, in 'Simulating Human Agents', AAAI Press, Technical report FS-00–03, pp. 88–96.

Matarić, M.J., Zordan, V. & Mason, Z. (1998), Movement Control Methods for Complex, Dynamically Simulated Agents: Adonis Dances the Macarena, in 'Proceedings, Autonomous Agents '98, Minneapolis/St. Paul, May 10–13', ACM Press, pp. 317–324.

Meltzoff, A. (1996), The human infant as imitative generalist: a 20-year progress report on infant imitation with implications for comparative psychology, in B.G. Galef & C.M. Heyes, eds, 'Social Learning in Animals: the Roots of Culture', Academic Press, New York, pp. 347–370.

Mitchell, R.W. (1987), A comparative-developmental approach to understanding imitation, in P.P.G. Bateson & P.H. Klopfer, eds, 'Perspectives in Ethology 7: Alternatives', Plenum Press, pp. 183–215.

Nadel, J. (2002), Imitation and imitation recognition: Functional use in preverbal infants and nonverbal children with autism, in W. Prinz & A.N. Meltzoff, eds, 'The Imitative Mind', Cambridge University Press, pp. 42–62.

Nadel, J. & Butterworth, G., eds (1999), *Imitation in Infancy*, Cambridge University Press.

Nadel, J., Guerini, C., Peze, A. & Rivet, C. (1999), The evolving nature of imitation as a format of communication, in J. Nadel & G. Butterworth, eds, 'Imitation in Infancy', Cambridge, pp. 209–234.

Nehaniv, C.L. (2003a), Evolvability and Interaction, in P. McOwan, K. Dautenhahn & C.L. Nehaniv, eds, 'Evolvability and Interaction: Evolutionary Substrates of Communication, Signaling, and Perception in the Dynamics of Social Complexity', EPSRC Network on Evolvability in Biological and Software Systems, proceedings available as University of Hertfordshire Computer Science Technical Report No.393.

Nehaniv, C.L. (2003b), Nine Billion Correspondence Problems and Some Methods for Solving Them, in 'Proc.Second International Symposium on Imitation in Animals and Artifacts — Aberystwyth, Wales, 7–11 April 2003', Society for the Study of Artificial Intelligence and Simulation of Behaviour, pp. 93–95.

Nehaniv, C. L. & Dautenhahn, K. (1998), Mapping between Dissimilar Bodies: Affordances and the Algebraic Foundations of Imitation, in J. Demiris & A. Birk, eds, 'Proceedings European Workshop on Learning Robots 1998 (EWLR-7), Edinburgh, 20 July 1998', pp. 64–72.

Nehaniv, C. L. & Dautenhahn, K. (2000), Of hummingbirds and helicopters: An algebraic framework for interdisciplinary studies of imitation and its applications, in J. Demiris & A. Birk, eds, 'Interdisciplinary Approaches to Robot Learning', World Scientific Series in Robotics and Intelligent Systems, pp. 136–161.

Nehaniv, C. L. & Dautenhahn, K. (2001), 'Like me? — Measures of correspondence and imitation', *Cybernetics and Systems* **32**(1–2), 11–51.

Nehaniv, C. L. & Dautenhahn, K. (2002), The Correspondence Problem, in K. Dautenhahn & C. L. Nehaniv, eds, 'Imitation in Animals and Artifacts', MIT Press, pp. 41–61.

Nielsen, M. & Dissanayake, C. (2003), Synchronic Imitation as Pre-linguistic Social Interaction, in 'Proc.Second International Symposium on Imitation in Animals and Artifacts — Aberystwyth, Wales, 7–11 April 2003', Society for the Study of Artificial Intelligence and Simulation of Behaviour, pp. 131–137.

Pepperberg, I. M. (2002), Allospecific Referential Speech Acquisition in Grey Parrots (*Psittacus erithacus*): Evidence for Multiple Levels of Avian Vocal Imitation, in K. Dautenhahn & C. L. Nehaniv, eds, 'Imitation in Animals and Artifacts', MIT Press, pp. 109–131.

Pfennig, D. W., Harcombe, W. R. & Pfenning, K. S. (2001), 'Frequency-dependent Batesian mimicry', *Nature* **410**, 323.

Rendell, L. & Whitehead, H. (2001), 'Culture in Whales and Dolphins', *Behavioral and Brain Sciences* **24**(2), 309–382.

Scassellati, B. (1999), Imitation and mechanisms of joint attention: A developmental structure for building social skills, in C. L. Nehaniv, ed., 'Computation for Metaphors, Analogy and Agents', Springer Lecture Notes in Artificial Intelligence, Volume 1562, pp. 176–195.

Schaal, S. (1999), 'Is imitation learning the route to humanoid robots?', *Trends in Cognitive Sciences* **3**(6), 233–242.

Thorndike, E. L. (1898), 'Animal intelligence: An experimental study of the associative process in animals', *Psychol. Rev. Monogr.* **2**, 551–553.

Thorpe, W. H. (1963), *Learning and Instinct in Animals*, London: Methuen.

Trevarthen, C., Kokkinaki, T. & Friamenghi, Jr., G. A. (1999), What infants' imitations communicate: With mothers, with fathers, and with peers, in J. Nadel & G. Butterworth, eds, 'Imitation in Infancy', Cambridge, pp. 128–185.

Visalberghi, E. & Fragaszy, D. (2002), Do Monkeys Ape? — Ten Years After, in K. Dautenhahn & C. L. Nehaniv, eds, 'Imitation in Animals and Artifacts', MIT Press.

Whiten, A. & Byrne, R. W. (1988), The Machiavellian intelligence hypotheses: editorial, in R. W. Byrne & A. Whiten, eds, 'Machiavellian intelligence', Clarendon Press, Chapter 1.

Whiten, A., Goodall, J., McGrew, W., Nishida, T., Reynolds, V., Sugiyama, Y., Tutin, C., Wrangham, R. & Boesch, C. (1999), 'Cultures in chimpanzees', *Nature* **399**, 682–685.

Zentall, T. R. (2001), 'Imitation in Animals: Evidence, Function and Mechanisms', *Cybernetics and Systems* **32**(1–2), 53–96.

Appendix

A. Metrics

The imitating agents can perceive the actions, states and effects of the model agents, and also their own actions, states and effects, and therefore require several error metrics to evaluate the similarity between each of them. Ideally the error metric value should be zero, indicating a perfect match. Note that our computational test-bed greatly simplifies the problem of how an imitator can perceive actions, states and effects of another agent. This simplification was necessary in order to focus on our particular research questions. However, if applied to physical robots, the perception of another agent's states, actions, and effects on the environment all represent very hard research problems.

All metrics described below can be used not only for comparisons between different agents, but also to evaluate the similarity between two actions, states or effects of the same agent. Notation is as in Figure 3 and Section 6.

The *state metric* calculates the averaged distance between the various joints of an agent (posed in a particular state) and the corresponding joints of another agent (posed in a different state) as if they were occupying the same workspace. Ideally this distance should be zero when the arms take corresponding poses, but this may not be possible due to embodiment differences. Using forward kinematics, the coordinates of the ends for each joint are found:

$$x_i = \sum_{j=1}^{i-1} x_j + l_i \cos\left(\sum_{j=1}^{i} \sigma_j \right) \tag{1a}$$

$$y_i = \sum_{j=1}^{i-1} y_j + l_i \sin\left(\sum_{j=1}^{i} \sigma_j \right) \tag{1b}$$

If both agents have the same number of joints the correspondence between them is straightforward; the Euclidean distance for each pair is calculated, the distances are then all summed and divided by the number of joints to give the metric value:

$$d_i = \sqrt{(x_i^{\text{model}} - x_i^{\text{imitator}})^2 + (y_i^{\text{model}} - y_i^{\text{imitator}})^2} \tag{2}$$

$$\mu^{\text{state}} = \frac{1}{n} \sum_{i=1}^{n} d_i \tag{3}$$

If the agents have a different number of joints, then some of the joints of the agent with the higher number of joints are ignored. In order to identify which joint corresponds with which, the *ratio* of the larger over the smaller number of joints is calculated, and if not an integer, is approximated by the nearest one. The ith joint of the agent with the smaller number of joints will correspond to the ($ratio \times i$)th joint of the agent with the larger number of joints. For example, if one of the agents has twice the number of joints, only every second joint will be considered.

The *action metric* uses the same algorithm as the state metric, but considers the *action* vectors instead of the *state* vectors.[10]

The value in the case of the state metric represents an absolute positional error; for the action metric, it represents the relative error between the change of the state angles, caused by the compared actions.

The *effect metric* is defined as the Euclidean length of the vector difference between two effects (x_1, y_1) and (x_2, y_2).

$$\mu^{\text{effect}} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{4}$$

If more than one aspect of the model behavior is used by the imitator agent, then a more complex type of metric can be considered, composed as a weighted combination of the different metric values for the aspects used. For example if the imitator attempts to match the model's actions and states, then the metric value could be a weighted sum of one-third the value of the action metric plus two-thirds the value of the state metric.

B. Granularity and the size of action and percept spaces

The size of the action space is ($\Pi_{i=1}^{N} c_i$), where N are the degrees of freedom of the agent (number of joints), and c_i ($1 \leq i \leq N$) denotes the number of possible values for the ith joint angle. Restricting c_i to only three possible values (+10°, 0° or −10°) results in a reasonably-sized[11] action space.

In some cases (depending on the particular agent embodiments), the most appropriate correspondence for a single model action might be a *sequence* of imitator actions. Allowing to consider sequences of actions (instead of single actions) increases the search space to ($\Pi_{i=1}^{N} c_i)^{B}$, where B is the number of actions within the sequence (assumed to be a constant).

This increase in search space size is not necessarily accompanied by a similar increase in the number of corresponding solutions. If the agents share a similar context (current state) and if the range of the possible angle values is restricted (as above), single actions can produce a comparable change of states and effects on the environment. In that setting, for two arbitrary agent embodiments, only a small fraction of action sequences will have an equal or better correspondence value than single actions.

Therefore in the current version of *Rabit*, a design choice was made to use only single imitator actions to correspond to single model actions, and this level of granularity (similar to the *trajectory-level* granularity used in *Chessworld*, see (Alissandrakis, Nehaniv & Dautenhahn 2002)) was used in the experiments presented in this paper. This choice introduces a limitation *by design* in the generating mechanism for proposing matching actions. A *key* to the correspondence library consists of states/action/effect, entry fields, encoding percepts, and possibly a proprioceptive entry field. The number K of such possible keys partially determines the size of the search space in solving the correspondence problem. In general if there are N degrees of freedom in the imitator, and c_i $(1 \leq i \leq N)$ denotes the number of possible choices of the action component for the ith degree of freedom, then $(\Pi_{i=1}^{N} c_i)^K$ is the size of the search space for the correspondence problem at the granularity of single actions. In our case N is the number of joints and $c_i = 3$ holds for all i, so we have a search space of size 3^{NK}.

Authors' addresses

Adaptive Systems Research Group
Faculty of Engineering & Information Sciences
University of Hertfordshire
Hatfield, Herts AL10 9AB, United Kingdom
A.Alissandrakis, C.L.Nehaniv, K.Dautenhahn@herts.ac.uk
http://adapsys.feis.herts.ac.uk/

About the authors

Aris Alissandrakis received his M.Eng. degree in Cybernetics from the Department of Cybernetics, University of Reading, Reading, U.K., in 1999. Since then he has pursued the Ph.D. degree at the Adaptive Systems Research Group, Department of Computer Science, University of Hertfordshire, U.K. In 2003 he submitted his Ph.D. thesis entitled "Imitation and Solving the Correspondence Problem for Dissimilar Embodiments — A Generic Framework". His research interests include artificial life and imitation in artificial systems. From 2004 he will be working as a postdoctoral Research Fellow in the European Integrated Project Cogniron, studying imitation in the context of robot-human interaction.

Chrystopher L. Nehaniv received his Ph.D. degree in Mathematics from the University of California, Berkeley, in 1992. He is currently Professor of Mathematical and Evolutionary Computer Sciences at the University of Hertfordshire, Hatfield, U.K. He has taught, lectured, and conducted interdisciplinary research at various universities in the United States, Japan, Great Britain, and Hungary. He is Associate Editor of the journal *BioSystems* and Director of the U.K. Engineering and Physical Sciences Research Council Network on Evolvability in Biological and Software Systems.

Kerstin Dautenhahn received her Ph.D. degree from the Biological Cybernetics Department of University of Bielefeld, Bielefeld, Germany, in 1993. She is currently Professor of Artificial Intelligence with the Adaptive Systems Research Group at the University of Hertfordshire in England. She previously worked at GMD, St. Augustin, Germany, and the VUB AI-Lab, Brussels, Belgium. Since 1993, she has initiated and led research projects on socially intelligent agents and social robotics, including a project of studying the use of robots in autism therapy. She is currently involved in several European projects on social agents and social robots (Victec, Elvis, Cogniron), and directs the Robotics and Interactive Systems Laboratory at the University of Hertfordshire, Hatfield, U.K. She is Editor-in-Chief, jointly with Harold Gouzoules, of the journal *Interaction Studies — Social Behaviour and Communication in Biological and Artificial Systems.*